Computer Model Calibration using High Dimensional Output

Dave Higdon, Los Alamos National Laboratory

Jim Gattiker, Los Alamos National Laboratory

Brian Williams, Los Alamos National Laboratory

Maria Rightley, Los Alamos National Laboratory

LA-UR-05-6410

This work focuses on combining observations from field experiments with detailed computer simulations of a physical process to carry out statistical inference. Of particular interest here is determining uncertainty in resulting predictions. This typically involves calibration of parameters in the computer simulator as well as accounting for inadequate physics in the simulator. We consider applications in characterizing material properties for which the field data and the simulator output are highly multivariate. For example, the experimental data and simulation output may be an image or may describe the shape of a physical object. We make use of the basic framework of Kennedy and O'Hagan (2001). However, the size and multivariate nature of the data lead to computational challenges in implementing the framework. To overcome these challenges, we make use of basis representations (e.g. principal components) to reduce the dimensionality of the problem and speed up the computations required for exploring the posterior distribution. This methodology is applied to applications, both ongoing and historical, at Los Alamos National Laboratory.

Keywords: computer experiments; predictability; uncertainty quantification; Gaussian process; predictive science; functional data analysis

1   Introduction

Understanding and predicting the behavior of complex physical processes is crucial in a variety of applications that include weather forecasting, oil reservoir management, hydrology, and impact dynamics. Investigations of such systems often make use of computer code–a simulator–which simulates the physical process of interest along with field data collected from experiments or observations of the actual physical system. The simulators we work with at Los Alamos National Laboratory (LANL) typically model fairly well-understood physical processes–this can be contrasted with agent-based simulations, which model social activity. Even so, uncertainties play an important role

in using the code to predict behavior of the physical system. Uncertainties arise from a variety of sources that include: uncertainty in the specification of initial conditions; uncertainty in the value of important physical constants (e.g. melting temperatures, equations of state, stress-strain rates); inadequate mathematical models in the code to describe physical behavior; and inadequacies in the numerical algorithms used for solving the specified mathematical systems (e.g. unresolved grids).

The features above clearly distinguish the simulation code from the actual physical system of interest. However much of this uncertainty can be mitigated by utilizing experimental observations to constrain uncertainties within the simulator. When the simulation code is sufficiently fast, estimation approaches based on Monte Carlo can be used. Examples include the Bayesain analysis of inverse problems via Markov chain Monte Carlo (MCMC) (Higdon *et al.*, 2003; Kaipio and Somersalo, 2004) or importance sampling (Berliner, 2001; Hegstad and Omre 2001). Such approaches offer the bennefit of readily accommodating very large parameter spaces in the estimation process. A limitation of these approaches is their reliance on the availability of a very fast simulation code. In some cases, the actual simulation code can be replaced by a local linear approximation so that MCMC may be employed; see Hanson and Cunningham (1999), Kaipio *et al.*(2000) or Nakhleh *et al.*(2005) for examples.

In applications such as weather forecasting where both the data and simulations arrive in a sequential fashion, filtering approaches can also be quite useful (Bengtsson *et al.*, 2003; Kao *et al.*, 2004). The sequential nature of such applications means that complex simulations need only be run over short-time intervals. Because of this, simulations are then more easily manipulated to better match the data as it arrives via Bayesian updating or other data assimilation techniques.

When the application is not readily amenable to sequential updating and the simulation code takes minutes–or even days–to complete, alternative estimation approaches are required. This is the case for our applications. In this paper we use the approach described in Kennedy and O'Hagan (2001) which utilizes the Gaussian process (GP) models described in Sacks *et al.*(1989) to model the simulator output at untried input settings. This model for the simulator is then embedded in a larger framework so that parameter estimation (i.e. calibration) and prediction can be carried out.

The method described in Kennedy and O'Hagan (2001) is appropriate when a single, univariate output is required from the simulator. In the applications we are currently involved in, the response of interest is often highly multivariate. Examples may include a one-dimensional displacement trace over time, or a time-sequence of two-dimensional images. To facilitate the description of our methodology, we utilize an application from the beginnings of the Manhattan project at LANL (Neddermeyer, 1943) in which steel cylinders were imploded by a high explosive (HE) charge

surrounding the cylinder. Figure 1 shows the results of such experiments. To describe these

Fig. 12.   No. 2:   3" OD, ½" wall, 8" long
                       TNT ¾" thick, 6" long

        No. 10:   Same except TNT 1½" thick, 7½" long
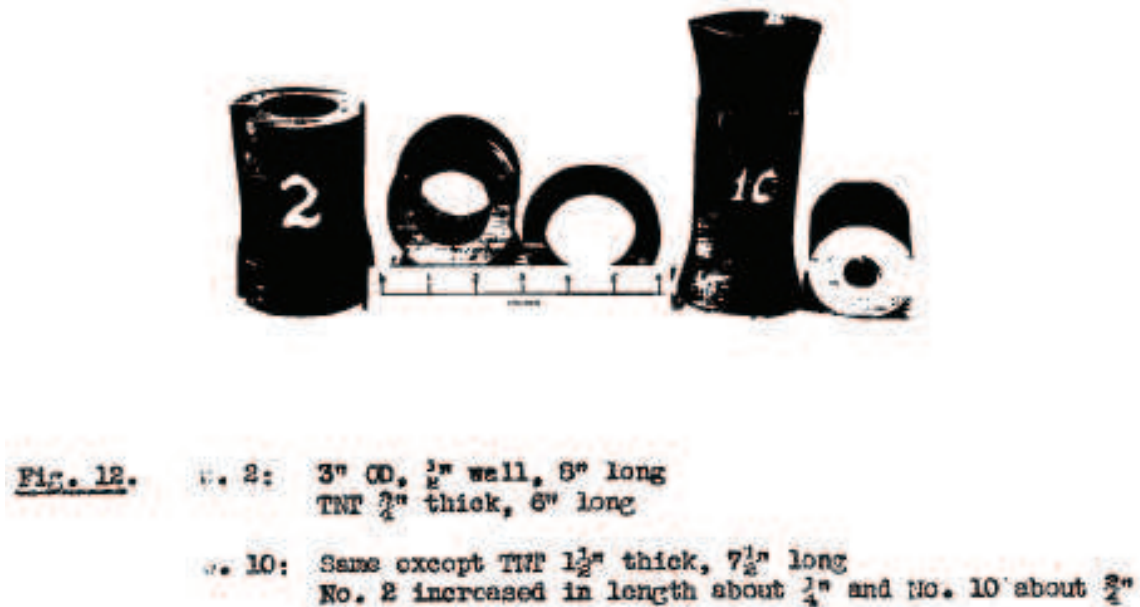   No. 2 increased in length about ¼" and No. 10 about ⅜"

Figure 1: Cylinders before and after implosion using TNT. Photos are from Neddermeyer (1943) experiments 2 and 10.

implosions, Neddermeyer devised a rudimentary physical model to simulate an experiment which depends on three inputs:

- x_1: the mass ratio between the HE and steel;

- θ_1: detonation energy per unit mass of HE;

- θ_2: yield stress of steel.

The first input x_1 is a condition under the control of the experimenter; the remaining two inputs $\theta = (\theta_1, \theta_2)$ are parameters that are to be estimated from the experimental data. More generally in the framework, we describe simulation inputs with the joint parameter (x, θ) where the p_x-vector x denotes controllable–or at least observable–input conditions of the experiment, and the p_θ-vector θ holds parameters to be calibrated, or estimated. Hence, a given simulation is controlled by a $(p_x + p_\theta)$-vector (x, θ), which contains the input settings. In this cylinder example we have $p_x = 1$ and $p_\theta = 2$.

Output from Neddermeyer's simulation model for a particular input setting (x^*, θ^*) is shown in Fig. 2. While this particular simulation code runs very quickly, we mean it to be a placeholder

for a more complicated, and computationally demanding, code from which a limited number of simmulations (typically less than 200) will be available for the eventual analysis.
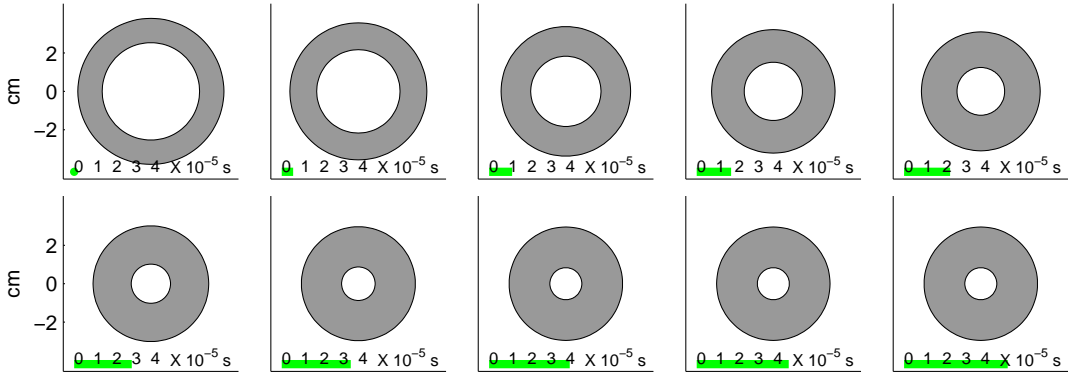

Figure 2: Evolution of the implosion of the steel cylinder using Neddermeyer's simulation model.

The data from this application come in the form of a sequence of high-speed photographs taken during the implosion, which takes place over a span of about 30 microseconds. The original photographs from the experiments were unavailable so we construct synthetic data using the rudimentary simulation model at assumed values for the calibration parameters θ. We generated data from three experiments, each with different values for x_1, the relative mass of HE to steel. The experimental data are shown in Fig. 3. As is typical of experiments we are involved in, the amount and condition of the observed data varies with experiment. Here the number of photographs and their timing varies with experiment. We take a trace of the inner radius of the cylinder to be the response of interest. The trace, described by angle ϕ and radius r, consists of 12 points equally spaced by angle.

We choose this example as the context to explain our approach because it possesses the features of the applications we are interested in, while still remaining quite simple. Specifically, we point the reader to the following properties of this example:

- The application involves a simulator from which only a limited number of simulations m ($m < 200$) may be carried out. Simulator output depends on a vector of input values (x, θ) where the p_x-vector x denotes the input specifications and the p_θ-vector θ holds parameters to be calibrated.

- The dimensionality of the input vector (x, θ) is limited. Here it is a three-dimensional vector ($p_x + p_\theta = 1 + 2 = 3$); in the application of Sec. 3 $p_\theta = 8$. We've worked with applications for which the dimension of (x, θ) is as large as 25. However applications with high-dimensional
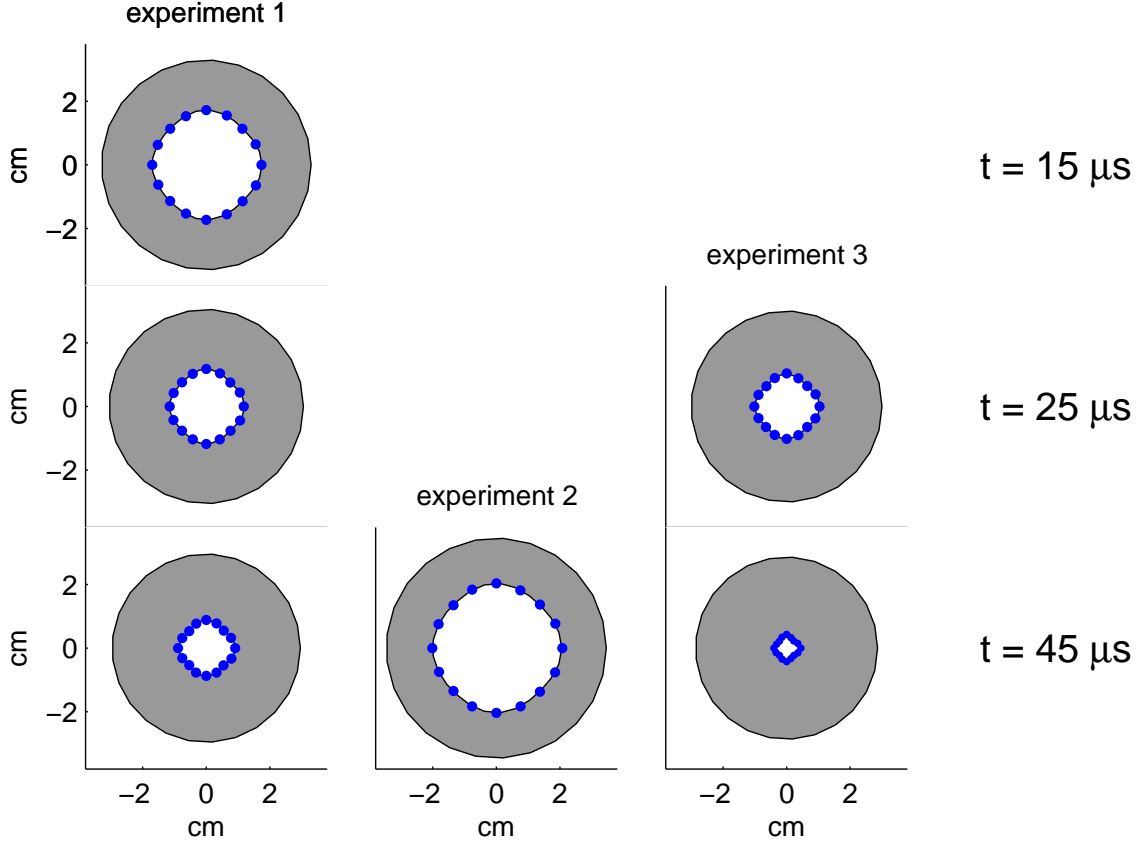
4

Figure 3: Hypothetical data obtained from photos at different times during the 3 experimental implosions. All cylinders initially had a 1.5 in outer and a 1.0 in inner radius.

$(p_\theta > 100)$ inputs, as is the case in inverse problems and tomography applications, are beyond the scope of the approach given here.

- Observations from one or more experiments are available to constrain the uncertainty regarding the calibration parameters θ. In most applications we've worked with, the number of experiments n is small $(n < 10)$. The experimental data are typically collected at various input conditions x, and the simulator produces output that is directly comparable to these field observations. Note that the simulator can also model the observation process used in the experiment to ensure that the simulations are compatable with the experimental data.

The goal of the analysis described in the next section is to:

- use the experimental data to constrain the uncertainty regarding the calibration parameters;

5

- make predictions (with uncertainty) at new input conditions x; and

- estimate systematic discrepancies between the simulator and physical reality.

We develop our methodology in the context of Neddermeyer's implosion application. However, the methodology generalizes to other applications in an obvious manner. Where generalizations are less obvious, we do our best to point this out. After describing the model formulation and posterior exploration via MCMC, we then illustrate this approach on an application in material science. The paper then concludes with a discussion.

2   Model Formulation

2.1   Design of simulator runs

A sequence of simulation runs is carried out at m input settings varying over predefined ranges for each of the input variables:

$$
\begin{pmatrix} x_1^* & \theta_1^* \\ \vdots & \vdots \\ x_m^* & \theta_m^* \end{pmatrix} = \begin{pmatrix} x_{11}^* & \cdots & x_{1p_x}^* & \theta_{11}^* & \cdots & \theta_{1p_\theta}^* \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{m1}^* & \cdots & x_{mp_x}^* & \theta_{m1}^* & \cdots & \theta_{mp_\theta}^* \end{pmatrix}. \tag{1}
$$

We would like to use this collection of simulation runs to screen inputs as well as to build simulator predictions at untried input settings using a Gaussian process model.

We typically use an orthogonal array (OA)-based Latin hypercube (LH) design (Tang, 1993) for the simulation runs for the applications we deal with. Such designs ensure space filling properties in higher dimensional margins, while maintaining the benefits of the LH design. We standardize the inputs to range over $[0, 1]^{p_x + p_\theta}$ to facilitate the design and prior specification (described later). Specifically, for the cylinder application, we use a strength 2, OA-based LH design for the simulation runs. The OA design is over $p_x + p_\theta = 1 + 2 = 3$ factors, with each factor having four levels equally spaced between 0 and 1: $\frac{1}{8}$, $\frac{3}{8}$, $\frac{5}{8}$, $\frac{7}{8}$. This $m = 36$ point design is shown in Fig. 4, along with the resulting OA-based LH design. We have found that in practice, this design of simulation runs is often built up sequentially.

For the cylinder application, the output from the resulting simulator runs is shown in Fig. 5. The simulator gives the radius of the inner shell of the cylinder over a fixed grid of times and angle. Surfaces from the left-hand frame are the output of three different simulations. Due to the symmetry assumptions in the simulator, the simulated inner radius only changes with time t–not
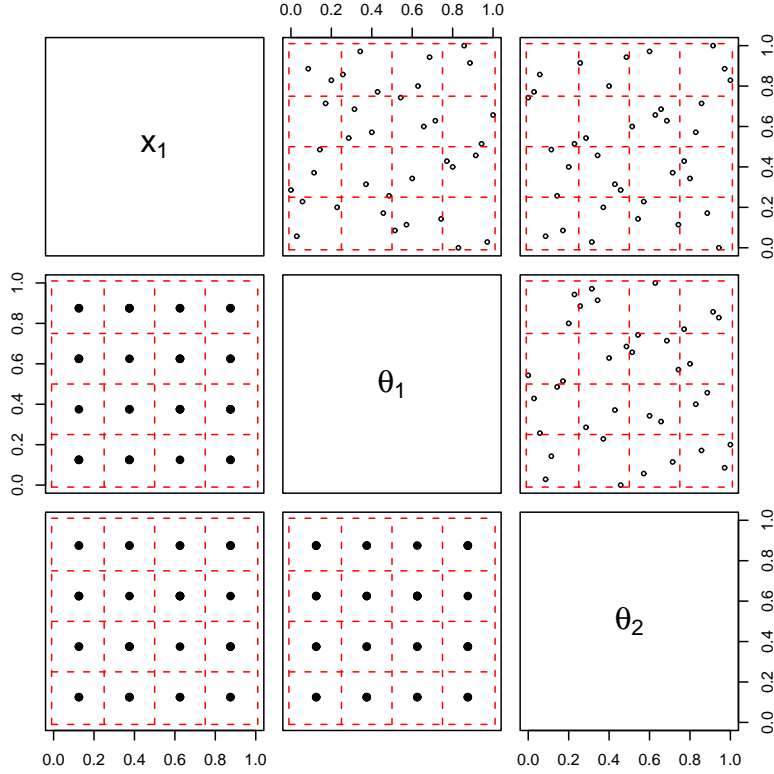
Figure 4: Lower triangle of plots: 2-dimensional projections of a $m = 36$ point, 4-level, OA design. Upper triangle of plots: An OA-based LH design obtained by spreading out the 4 level OA design so that each 1-dimensional projection gives an equally spaced set of points along [0,1].

angle ϕ. However, since the experimental data give radius values that vary by angle at fixed sets of times (Fig. 3), we treat the simulator output as an image of radii over time t and angle ϕ. All $m = 36$ simulations are shown in the right frame of Fig. 5 as a function of time only. It is worth noting that the simulations always give the output on this fixed grid over time and angle. This is in contrast to the comparatively irregular collection of experimental data that varies substantially as to its amount as well as which angles and times the radius is measured.
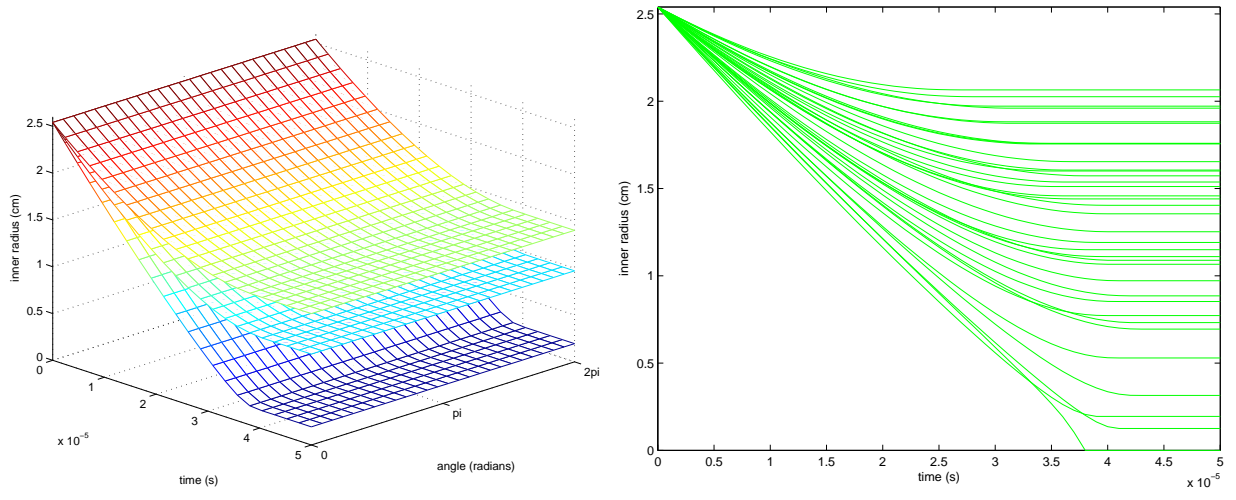
Figure 5: Simulated implosions using input settings from the OA-based LH design. Simulation output gives radius by time (t) and angle ϕ as shown in the left-hand figure. The radius by time trajectories are shown for all $m = 36$ simulations in the right-hand figure.

2.2 Modeling simulator output

Our analysis requires we develop a probability model to describe the simulator output at untried settings (x, θ). To do this, we use the simulator outputs to construct a GP model that "emulates" the simulator at arbitrary input settings over the (standardized) design space $[0, 1]^{p_x + p_\theta}$. To construct this emulator, we model the simulation output using a p_η-dimensional basis representation:

$$\eta(x, \theta) = \sum_{i=1}^{p_\eta} k_i w_i(x, \theta) + \epsilon, \ (x, \theta) \in [0, 1]^{p_x + p_\theta}, \tag{2}$$

where $\{k_1, \ldots, k_{p_\eta}\}$ is a collection of orthogonal, n_η-dimensional basis vectors, the $w_i(x, \theta)$'s are GPs over the input space, and ϵ is a n_η-dimensional error term. This type of formulation reduces the problem of building an emulator that maps $[0, 1]^{p_x + p_\theta}$ to R^{n_η} to building p_η independent, univariate GP models for each $w_i(x, \theta)$. The details of this model specification are given below.

Output from each of the m simulation runs prescribed by the design results in n_η-dimensional vectors, which we denote by $\eta_1, \ldots, \eta_m$. Since the simulations rarely give incomplete output, the simulation output can often be efficiently represented via principal components (Ramsay and Silverman, 1997). We first standardize the simulations by centering the raw simulation output vectors about the mean of these vectors: $\frac{1}{m} \sum_{j=1}^{m} \eta_j$. We then scale the output by a single value so that its variance is 1. This standardization simplifies some of the prior specifications in our models. We also note that, depending on the application, some alternative standardization may be preferred. Whatever the choice of the standardization, the same standardization is also applied to the experimental data.

We define Ξ to be the $n_\eta \times m$ matrix obtained by column-binding the (standardized) output vectors from the simulations

$$\Xi = [\eta_1; \cdots; \eta_m].$$

Typically, the size of a given simulation output n_η is much larger than the number of simulations carried out m. We apply the singular value decomposition (SVD) to the simulation output matrix Ξ giving

$$\Xi = UDV^T,$$

where U is a $n_\eta \times m$ matrix having orthogonal columns, D is a diagonal $m \times m$ matrix holding the singular values, and V is a $m \times m$ orthonormal matrix. To construct a p_η-dimensional representation of the simulation output, we define the principal component (PC) basis matrix K_η to be the first p_η columns of $[\frac{1}{\sqrt{m}} UD]$. The resulting principal component loadings or weights is then given by $[\sqrt{m}V]$, whose columns have mean 0 and variance 1.

For the cylinder example we take $p_\eta = 3$ so that $K_\eta = [k_1; k_2; k_3]$; the basis functions k_1, k_2 and k_3 are shown in Fig. 6. Note that the k_i's do not change with ϕ due to the angular invariance of
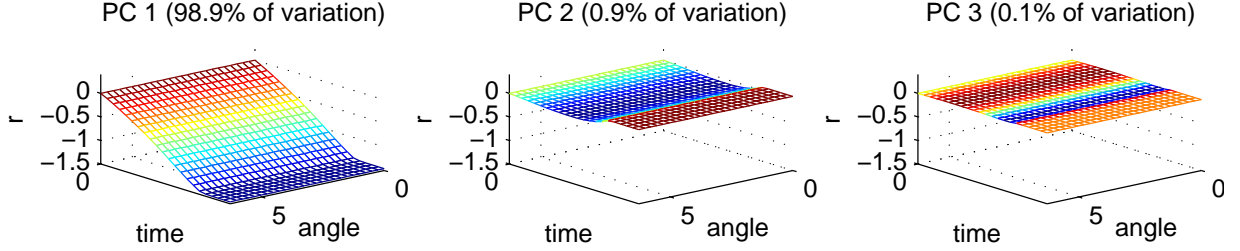

Figure 6: Principal component bases derrived from the simulation output.

Neddermeyer's simulation model.

We use the basis representation of Eq. (2) to model the n_η-dimensional simulator output over the input space. Each PC weight $w_i(x,\theta)$, $i = 1, \ldots, p_\eta$, is then modeled as a mean 0 GP

$$w_i(x,\theta) \sim \text{GP}(0, \lambda_{wi}^{-1} R((x,\theta),(x',\theta'); \rho_{wi})),$$

where λ_{wi} is the marginal precision of the ith process and the correlation function is given by

$$R((x,\theta),(x',\theta'); \rho_{wi}) = \prod_{k=1}^{p_x} \rho_{wik}^{4(x_k-x'_k)^2} \times \prod_{k=1}^{p_\theta} \rho_{wi(k+p_x)}^{4(\theta_k-\theta'_k)^2}. \tag{3}$$

This is the Gaussian covariance function, which gives very smooth realizations, and has been used previously by Sacks *et al.*(1989) and Kennedy and O'Hagan (2001) to model computer simulation output. An advantage of this product form is that only a single additional parameter is required per additional input dimension, while the fitted GP response still allows for rather general interactions between inputs. We use this Gaussian form for covariance function because the simulators we work with tend to respond very smoothly to changes in the inputs. Depending on the nature of the sensitivity of simulation output to input changes, one may wish to alter this covariance specification to allow for rougher realizations. The parameter ρ_{wik} controls the spatial range for the kth input dimension of the process $w_i(x,\theta)$. Under this parameterization, ρ_{wik} gives the correlation between $w_i(x,\theta)$ and $w_i(x',\theta')$ when the input conditions (x,θ) and (x',θ') are identical, except for a difference of 0.5 in input dimension k. Note that this interpretation makes use of the standardization of the input space to $[0,1]^{p_x+p_\theta}$.

Restricting to the m input design settings given in Eq. (1), we define the m-vector w_i to be the

10

restriction of the process $w_i(\cdot, \cdot)$ to the input design settings

$$w_i = (w_i(x_1^*, \theta_1^*), \ldots, w_i(x_m^*, \theta_m^*))^T, \quad i = 1, \ldots, p_\eta.$$

In addition we define $R((x^*, \theta^*); \rho_{wi})$ to be the $m \times m$ correlation matrix resulting from applying (3) to each pair of input settings in the design. The $(p_x + p_\theta)$-vector ρ_{wi} gives the correlation distances for each of the input dimensions.

At the m simulation input settings, the mp_η-vector $w = (w_1^T, \ldots, w_{p_\eta}^T)^T$ then has prior disribution

$$w = \begin{pmatrix} w_1 \\ \vdots \\ w_{p_\eta} \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} \lambda_{w1}^{-1} R((x^*, \theta^*); \rho_{w1}) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_{wp_\eta}^{-1} R((x^*, \theta^*); \rho_{wp_\eta}) \end{pmatrix} \right), \quad (4)$$

which is controlled by p_η precision parameters held in λ_w and $p_\eta \cdot (p_x + p_\theta)$ spatial correlation parameters held in ρ_w. The centering of the simulation output makes the zero mean prior appropriate. The prior above can be written more compactly as

$$w \sim N(0, \Sigma_w),$$

where Σ_w, controlled by parameter vectors λ_w and ρ_w, is given in (4).

We specify independent $\Gamma(a_w, b_w)$ priors for each λ_{wi} and independent $\text{beta}(a_{\rho_w}, b_{\rho_w})$ priors for the ρ_{wik}'s.

$$\pi(\lambda_{wi}) \propto \lambda_{wi}^{a_w - 1} e^{-b_w \lambda_{wi}}, \quad i = 1, \ldots, p_\eta,$$

$$\pi(\rho_{wik}) \propto \rho_{wik}^{a_{\rho_w} - 1} (1 - \rho_{wik})^{b_{\rho_w} - 1}, \quad i = 1, \ldots, p_\eta, \; k = 1, \ldots, p_x + p_\theta.$$

We expect the marginal variance for each $w_i(\cdot, \cdot)$ process to be close to one due to the standardization of the simulator output. For this reason we specify that $a_w = b_w = 5$. In addition, this informative prior helps stabilize the resulting posterior distribution for the correlation parameters which can trade off with the marginal precision parameter (Kern (2000)).

Because we expect only a subset of the inputs to influence the simulator response, our prior for the correlation parameters reflects this expectation of "effect sparsity." Under the parameterization in (3), input k is inactive for PC i if $\rho_{wik} = 1$. Choosing $a_{\rho_w} = 1$ and $0 < b_{\rho_w} < 1$ will give a density with substantial prior mass near 1. For the cylinder example we take $b_{\rho_w} = 0.1$, which makes $\Pr(\rho_{wik} < 0.98) \approx \frac{1}{3}$ a priori. In general, the selection of these hyperparameters should depend on how many of the $p_x + p_\theta$ inputs are expected to be active.

If we take the error vector in the basis representation of (2) to be i.i.d. normal, we can then develop the sampling model, or likelihood, for the simulator output. We define the mn_η-vector η to be the concatenation of all m simulation output vectors

$$\eta = \text{vec}(\Xi) = \text{vec}([\eta(x_1^*, \theta_1^*); \cdots ; \eta(x_m^*, \theta_m^*)]).$$

Given precision λ_η of the errors the likelihood is then

$$L(\eta|w, \lambda_\eta) \propto \lambda_\eta^{\frac{mn_\eta}{2}} \exp\left\{-\tfrac{1}{2}\lambda_\eta(\eta - Kw)^T(\eta - Kw)\right\},$$

where the $mn_\eta \times mp_\eta$ matrix K is given by

$$K = [I_m \otimes k_1; \cdots ; I_m \otimes k_{p_\eta}],$$

and the k_i's are the p_η basis vectors previously computed via SVD. A $\Gamma(a_\eta, b_\eta)$ is specified for the error precision λ_η.

Since the likelihood factors as shown below

$$L(\eta|w, \lambda_\eta) \propto \lambda_\eta^{\frac{mp_\eta}{2}} \exp\left\{-\tfrac{1}{2}\lambda_\eta(w - \hat{w})^T(K^TK)(w - \hat{w})\right\} \times$$
$$\lambda_\eta^{\frac{m(n_\eta - p_\eta)}{2}} \exp\left\{-\tfrac{1}{2}\lambda_\eta \eta^T(I - K(K^TK)^{-1}K^T)\eta\right\},$$

the formulation can be equivalently represented with a dimension reduced likelihood and a modified $\Gamma(a_\eta', b_\eta')$ prior for λ_η:

$$L(\hat{w}|w, \lambda_\eta) \propto \lambda_\eta^{\frac{mp_\eta}{2}} \exp\left\{-\tfrac{1}{2}\lambda_\eta(\hat{w} - w)^T(K^TK)(\hat{w} - w)\right\},$$

where

$$
\begin{aligned}
a_\eta' &= a_\eta + \frac{m(n_\eta - p_\eta)}{2}, \\
b_\eta' &= b_\eta + \tfrac{1}{2}\eta^T(I - K(K^TK)^{-1}K^T)\eta, \text{ and} \\
\hat{w} &= (K^TK)^{-1}K^T\eta.
\end{aligned}
\tag{5}
$$

Thus the normal-gamma model

$$\eta|w, \lambda_\eta \sim N(Kw, \lambda_\eta^{-1} I_{mn_\eta}), \quad \lambda_\eta \sim \Gamma(a_\eta, b_\eta)$$

is equivalent to the reduced form

$$\hat{w}|w, \lambda_\eta \sim N(w, (\lambda_\eta K^TK)^{-1}), \quad \lambda_\eta \sim \Gamma(a_\eta', b_\eta')$$

since

$$L(\eta|w, \lambda_\eta) \times \pi(\lambda_\eta; a_\eta, b_\eta) \propto L(\hat{w}|w, \lambda_\eta) \times \pi(\lambda_\eta; a'_\eta, b'_\eta). \tag{6}$$

The likelihood depends on the simulations only through the computed PC weights $\hat{w}$. After integrating out w, the posterior distribution becomes

$$
\begin{aligned}
\pi(\lambda_\eta, \lambda_w, \rho_w|\eta) \quad \propto \quad & \left|(\lambda_\eta K^T K)^{-1} + \Sigma_w\right|^{-\frac{1}{2}} \exp\{-\tfrac{1}{2}\hat{w}^T([\lambda_\eta K^T K]^{-1} + \Sigma_w)^{-1}\hat{w}\} \times \tag{7} \\
& \lambda_\eta^{a'_\eta - 1} e^{-b'_\eta \lambda_\eta} \times \prod_{i=1}^{p_\eta} \lambda_{wi}^{a_w - 1} e^{-b_w \lambda_{wi}} \times \\
& \prod_{i=1}^{p_\eta} \left\{ \prod_{j=1}^{p_x} (1 - \rho_{wij})^{b_\rho - 1} \prod_{j=1}^{p_\theta} (1 - \rho_{wi(j+p_x)})^{b_\rho - 1} \right\}.
\end{aligned}
$$

This posterior distribution is a milepost on the way to the complete formulation, which also incorporates experimental data. However, it is worth considering this intermediate posterior distribution for the simulator response. It can be explored via MCMC using standard Metropolis updates and we can view a number of posterior quantities to illuminate features of the simulator. Oakley and O'Hagan (2004) use the posterior of the simulator response to investigate formal sensitivity measures of a univariate simulator; Sacks *et al.*, (1989) do this from a non-Bayesian perspective. For example, Fig. 7 shows boxplots of the posterior distributions for the components of ρ_w. From this figure it is apparent that PC 1 is most influenced by x_1–the relative amount of HE used in the experiment.

Given the posterior realizations from (7), one can generate realizations from the process $\eta(x, \theta)$ at any input setting $(x^\star, \theta^\star)$. Since

$$\eta(x^\star, \theta^\star) = \sum_{i=1}^{p_\eta} k_i w_i(x^\star, \theta^\star),$$

realizations from the $w_i(x^\star, \theta^\star)$ processes need to be drawn given the MCMC output. For a given draw $(\lambda_\eta, \lambda_w, \rho_w)$ a draw of $w^\star = (w_1(x^\star, \theta^\star), \ldots, w_{p_\eta}(x^\star, \theta^\star))^T$ can be produced by making use of the fact

$$\begin{pmatrix} \hat{w} \\ w^\star \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \left[\begin{pmatrix} (\lambda_\eta K^T K)^{-1} & 0 \\ 0 & 0 \end{pmatrix} + \Sigma_{w,w^\star}(\lambda_w, \rho_w)\right]\right),$$

where $\Sigma_{w,w^\star}$ is obtained by applying the prior covariance rule to the augmented input settings that include the original design and the new input setting $(x^\star, \theta^\star)$. Recall $\hat{w}$ is defined in (5). Application of the conditional normal rules then gives

$$w^\star|\hat{w} \sim N(V_{21}V_{11}^{-1}\hat{w}, V_{22} - V_{21}V_{11}^{-1}V_{12}),$$
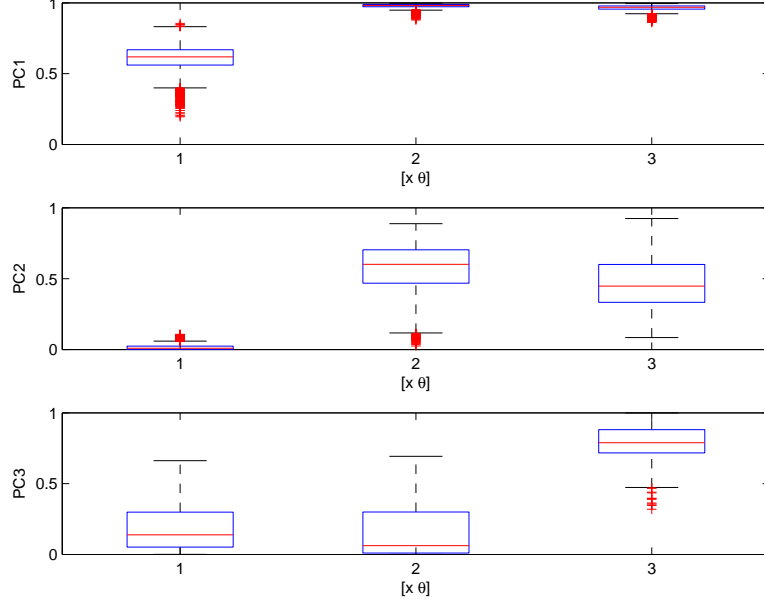
Figure 7: Boxplots of posterior samples for each ρ_{wik} for the cylinder application.

where

$$V = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} = \left[\begin{pmatrix} (\lambda_\eta K^T K)^{-1} & 0 \\ 0 & 0 \end{pmatrix} + \Sigma_{w,w^\star}(\lambda_w, \rho_w) \right]$$

is a function of the parameters produced by the MCMC output. Hence, for each posterior realization of $(\lambda_\eta, \lambda_w, \rho_w)$, a realization of $w^\star$ can be produced. The above recipe easily generalizes to give predictions over many input settings at once.

Figure 8 shows posterior means for the simulator response η where each of the three inputs were varied over their prior range of $[0, 1]$ while the other two inputs were held at their nominal setting of 0.5. The posterior mean response surfaces convey an idea of how the different parameters affect the highly multivariate simulation output. Other marginal functionals of the simulation response can also be calculated such as sensitivity indicies or estimates of the Sobol decomposition (Sacks et al., 1989; Oakley and O'Hagan, 2004).
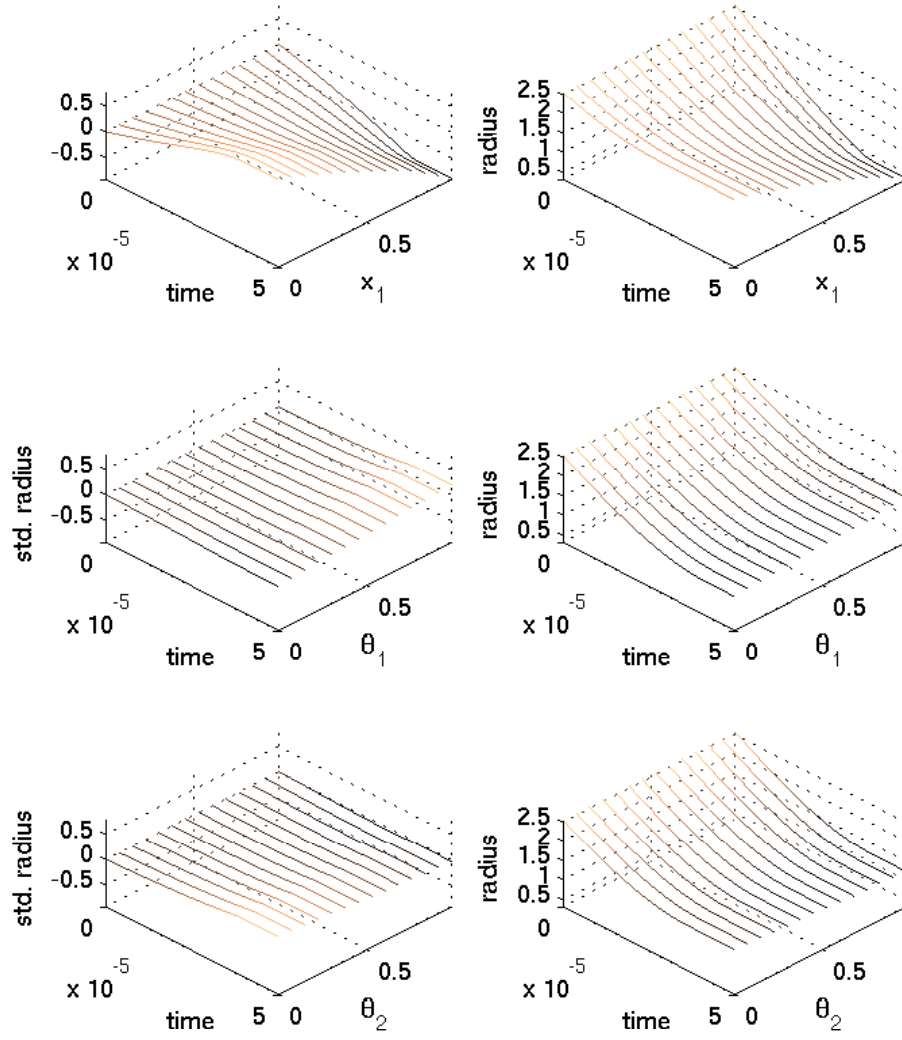
Figure 8: Posterior mean simulator predictions (radius as a function of time) varying 1 input, holding others at their nominal values of 0.5. Left-hand column shows predictions on the standardized scale; right-hand column shows the predictions on the original scale.

2.3 General model

The model for the simulator response is one component of the complete model formulation, which uses experimental data to calibrate the parameter vector θ as well as account for inadequacies in the simulator. We closely follow the formulation of Kennedy and O'Hagan (2001). Here a vector of experimental observations $y(x)$ taken at input condition x is modeled as

$$y(x) = \eta(x, \theta) + \delta(x) + \epsilon,$$

where $\eta(x, \theta)$ is the simulated output at the true parameter setting θ, $\delta(x)$ accounts for discrepancy between the simulator and physical reality, and ϵ models observation error. For discussion and motivation regarding this particular decomposition see Kennedy and O'Hagan (2001) and accompanying discussion.

2.4 Discrepancy model

Previously, Sec. 2.2 gave a detailed explanation of our GP model for $\eta(x, \theta)$. In this section we define the discrepancy model which, like the model for $\eta(x, \theta)$, is constructed using a basis representation, placing GP models on the basis weights. It differs in that the basis weights depend only on input condition x and that the basis specification for $\delta(x)$ is typically nonorthogonal and tailored to the application at hand.

For this example consisting of imploding of steel cylinders, $\delta(x)$ adjusts the simulated radius over the time $\times$ angle grid. This discrepancy between actual and simulated radius is constructed as a linear combination of $p_\delta = 24$ basis functions that are defined over the $n_\eta = 20 \times 26$ grid over time t and angle ϕ. Thus

$$\delta(x) = \sum_{k=1}^{p_\delta} d_k(t, \phi) v_k(x) = \sum_{k=1}^{p_\delta} d_k v_k(x), \tag{8}$$

where the basis functions d_k, $k = 1, \ldots, p_\delta$, are shown in Fig. 9, and independent GP priors over x are specified for each weight $v_k(x)$.

The basis functions are specified according to what is known about the actual physical process and potential deficiencies in the simulator. Here the basis functions are separable normal kernels that are long in the t direction and narrow and periodic in the ϕ direction. This conforms to our expectation that discrepancies–if they exist–should have a strong time persistence, with a much weaker angular persistence.
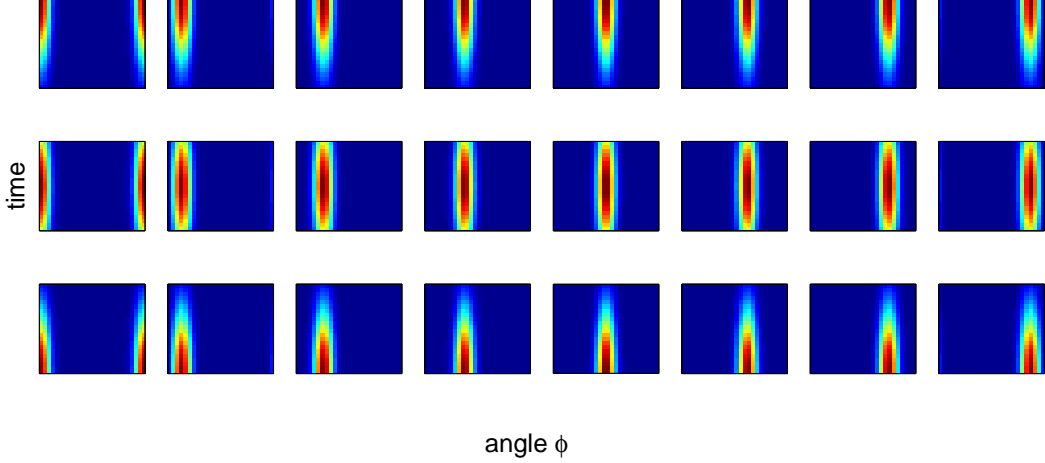
16

Figure 9: Basis kernels d_k, $k = 1, \ldots, p_\delta$. Each kernel is a $n_\eta = 20 \times 26$ image over time (y-axis) and angle (x-axis). Note that the basis elements are periodic over angle ϕ.

We specify independent mean 0 GP priors for each basis weight $v_k(x)$. Thus the p_δ-variate process $v(x) = (v_1(x), \ldots, v_{p_\delta}(x))^T$ is a mean 0 GP with covariance rule given by

$$\text{Cov}(v(x), v(x')) = \lambda_v^{-1} I_{p_\delta} \otimes R(x, x'; \rho_v),$$

where λ_v is the common marginal precision of each $v_k(x)$, ρ_v is a p_x-vector controlling the correlation strength along each component of x, and $R(x, x'; \rho_v)$ is a stationary Gaussian product correlation model given by

$$R(x, x'; \rho_v) = \prod_{k=1}^{p_x} \rho_{vk}^{4(x_k - x_k')^2}. \tag{9}$$

Note that the Gaussian form of the correlation will enforce a high degree of smoothness for each process $v_k(x)$ as a function of x. We feel this is plausible in this application since we expect any discrepancies to change smoothly with input condition x. Other applications may require an alternate specification.

As with the GP model for the simulator $\eta(x, \theta)$, we complete the discrepancy model formulation by specifying a gamma prior for the precision λ_v and independent beta priors for the components of ρ_v,

$$
\begin{aligned}
\pi(\lambda_v) &\propto \lambda_v^{a_v - 1} e^{-b_v \lambda_v} \\
\pi(\rho_{vk}) &\propto \rho_{vk}^{a_{\rho_v} - 1} (1 - \rho_{vk})^{b_{\rho_v} - 1}, \quad k = 1, \ldots, p_\delta,
\end{aligned}
$$

where $a_v = 1$, $b_v = 0.0001$, $a_{\rho_v} = 1$, and $b_{\rho_v} = 0.1$. This results in a rather uninformative prior for the precision λ_v. If the data are uninformative about this parameter, it will tend to stay at large

17

values that are consistent with a very small discrepancy. Like the prior for ρ_w, we take $a_{\rho_v} = 1$ and $b_{\rho_v} = 0.1$ to encourage effect sparsity.

2.5 Full model specification; priors and smoothness

Given the model specifications for the simulator $\eta(x, \theta)$ and the discrepancy $\delta(x)$, we can now consider the sampling model for the experimentally observed data. We assume the data $y(x_1), \ldots, y(x_n)$ are collected for n experiments at input conditions $x_1, \ldots, x_n$. For the implosion example, there are $n = 3$ experiments whose data are shown in Fig. 3. Each $y(x_i)$ is a collection of n_{y_i} measurements over points indexed by time and angle configurations $(t_{i1}, \phi_{i1}), \ldots, (t_{in_{y_i}}, \phi_{in_{y_i}})$. The data for experiment i is modeled as the sum of the simulator output at the true parameter setting θ and the discrepancy

$$y(x_i) \quad = \quad \eta(x_i, \theta) + \delta(x_i) + e_i,$$

where the observation error vector e_i is modeled as $N\left(0, (\lambda_y W_i)^{-1}\right)$. Using the basis representations for the simulator and the discrepancies, this becomes

$$y(x_i) \quad = \quad K_i w(x_i, \theta) + D_i v(x_i) + e_i.$$

Because the time $\times$ angle support of each $y(x_i)$ varies with experiment and isn't necessarily contained in the support of the simulation output, the basis vectors in K_i may have to be interpolated over time and angle from K_η. Since the simulation output over time and angle is quite dense, this interpolation is straightforward. The discrepancy basis matrix D_i is determined by the functional form given in (8)–the jk element of D_i is given by

$$D_{ijk} = d_k(t_{ij}, \phi_{ij}).$$

The sampling model for the observations in experiment i is n_{y_i}-variate normal

$$y(x_i) | w(x_i, \theta), v(x_i), \lambda_y \quad \sim \quad N\left([D_i; K_i] \begin{pmatrix} v(x_i) \\ w(x_i, \theta) \end{pmatrix}, (\lambda_y W_i)^{-1}\right).$$

Taking all of the experiments together, the sampling model is n_y variate normal, where $n_y = n_{y_1} + \cdots + n_{y_n}$, is the total number of experimental data points. We define y to be the n_y-vector from concatenation of the $y(x_i)$'s, $v = \text{vec}([v(x_1); \cdots; v(x_n)]^T)$ and $u(\theta) = \text{vec}([w(x_1, \theta); \cdots; w(x_n, \theta)]^T)$. The sampling model for the entire experimental dataset, along with the prior for the observation

18

precision λ_y, can be written as

$$y|v, u(\theta), \lambda_y \sim \mathrm{N}\left(B\begin{pmatrix} v \\ u(\theta) \end{pmatrix}, (\lambda_y W_y)^{-1}\right), \quad \lambda_y \sim \Gamma(a_y, b_y), \tag{10}$$

where $W_y = \mathrm{diag}(W_1, \ldots, W_n)$,

$$B = [\mathrm{diag}(D_1, \ldots, D_n); \mathrm{diag}(K_1, \ldots, K_n)]\begin{pmatrix} P_D^T & 0 \\ 0 & P_K^T \end{pmatrix},$$

and P_D and P_K are permutation matricies whose rows are given by

$$P_D(j + n(i-1); \cdot) = e_{(j-1)p_\delta + i}^T, \quad i = 1, \ldots, p_\delta; \; j = 1, \ldots, n$$

$$P_K(j + n(i-1); \cdot) = e_{(j-1)p_\eta + i}^T, \quad i = 1, \ldots, p_\eta; \; j = 1, \ldots, n.$$

Note that permutations are required for specifying B since the basis weight components v and $u(\theta)$ are separated in (10). The observation precision W_y is often fairly well-known in practice. Hence we use an informative prior for λ_y that encourages its value to be near one. In the implosion example we set $a_y = b_y = 5$.

Equivalently (10) can be represented using the normal-gamma form

$$\begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix}\Big| \begin{pmatrix} v \\ u(\theta) \end{pmatrix}, \lambda_y \sim \mathrm{N}\left(\begin{pmatrix} v \\ u(\theta) \end{pmatrix}, (\lambda_y B^T W_y B)^{-1}\right), \quad \lambda_y \sim \Gamma(a_y', b_y'),$$

with

$$n_y = n_{y_1} + \cdots + n_{y_n}, \text{ denoting the total number of experimental data points,}$$

$$\begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix} = (B^T W_y B)^{-1} B^T W_y y,$$

$$a_y' = a_y + \tfrac{1}{2}[n_y - n(p_\delta + p_\eta)], \text{ and}$$

$$b_y' = b_y + \frac{1}{2}\left[\left(y - B\begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix}\right)^T W_y \left(y - B\begin{pmatrix} \hat{v} \\ \hat{u} \end{pmatrix}\right)\right].$$

This equivalency follows from (6) given in Sec. 2.2.

The (marginal) distribution for the combined, reduced data obtained from the experiments and simulations given the covariance parameters has the form

$$\begin{pmatrix} \hat{v} \\ \hat{u} \\ \hat{w} \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Lambda_y^{-1} & & 0 \\ & & 0 \\ 0 & 0 & \Lambda_\eta^{-1} \end{pmatrix} + \begin{pmatrix} \Sigma_v & 0 & 0 \\ 0 & & \\ 0 & & \Sigma_{uw} \end{pmatrix}\right), \tag{11}$$

19

where

$$\Lambda_y = \lambda_y B^T W_y B,$$

$$\Lambda_\eta = \lambda_\eta K^T K,$$

$$\Sigma_v = \lambda_v^{-1} I_{p_\delta} \otimes R(x,x;\rho_v),$$

and the covariance matrix Σ_{uw}, which links the simulator response $u(\theta)$ at the experimental settings, (x_i, θ), $i = 1, \ldots, n$, to the simulator response w at the design inputs, (x_j^*, θ_j^*), $j = 1, \ldots, m$, is given by

$$
\begin{pmatrix}
\lambda_{w1}^{-1} R((x,\theta),(x,\theta);\rho_{w1}) & 0 & 0 & \lambda_{w1}^{-1} R((x,\theta),(x^*,\theta^*);\rho_{w1}) & 0 & 0 \\
0 & \ddots & 0 & 0 & \ddots & 0 \\
0 & 0 & \lambda_{wp_\eta}^{-1} R((x,\theta),(x,\theta);\rho_{wp_\eta}) & 0 & 0 & \lambda_{wp_\eta}^{-1} R((x,\theta),(x^*,\theta^*);\rho_{wp_\eta}) \\
\lambda_{w1}^{-1} R((x^*,\theta^*),(x,\theta);\rho_{w1}) & 0 & 0 & \lambda_{w1}^{-1} R((x^*,\theta^*),(x^*,\theta^*);\rho_{w1}) & 0 & 0 \\
0 & \ddots & 0 & 0 & \ddots & 0 \\
0 & 0 & \lambda_{wp_\eta}^{-1} R((x^*,\theta^*),(x,\theta);\rho_{wp_\eta}) & 0 & 0 & \lambda_{wp_\eta}^{-1} R((x^*,\theta^*),(x^*,\theta^*);\rho_{wp_\eta})
\end{pmatrix}.
$$

Above, $R(x, x; \rho_v)$ denotes the $n \times n$ correlation matrix for the discrepancy process obtained by applying (9) to the input conditions $x_1, \ldots, x_n$ corresponding to the n experiments; $R((x^*, \theta^*), (x, \theta); \rho_{wi})$ denotes the $m \times n$ correlation submatrix for the GP modeling the simulator output obtained by applying (3) to the m simulator input settings $(x_1^*, \theta_1^*), \ldots, (x_m^*, \theta_m^*)$ crossed with the n experimental settings $(x_1, \theta), \ldots, (x_n, \theta)$, with θ denoting the true, but unknown, calibration setting to be estimated. The remaining components of Σ_{uw} are constructed analogously. Note that only the off-diagonal blocks of Σ_{uw} depend on the unknown calibration parameters contained in θ. The equivalency of (6) reduces the $(n_y + mn_\eta)$-variate normal distribution of $(y^T, \eta^T)^T$ to the $(n(p_\eta + p_\delta) + mp_\eta)$-variate normal distribution of $(\hat{v}^T, \hat{u}^T, \hat{w}^T)^T$ given in (11)–particularly efficient when n_η and n_y are large.

2.5.1 Posterior distribution

If we take $\hat{z}$ to denote the reduced data $(\hat{v}^T, \hat{u}^T, \hat{w}^T)^T$, and $\Sigma_{\hat{z}}$ to be the covariance matrix given in (11), the posterior distribution has the form

$$\pi(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta | y, \eta) \propto \tag{12}$$

$$|\Sigma_{\hat{z}}|^{-\frac{1}{2}} \exp\left\{ -\tfrac{1}{2} \hat{z}^T \Sigma_{\hat{z}}^{-1} \hat{z} \right\} \times \lambda_\eta^{a_\eta' - 1} e^{-b_\eta' \lambda_\eta} \times \prod_{i=1}^{p_\eta} \lambda_{wi}^{a_w - 1} e^{-b_w \lambda_{wi}} \times$$

$$\prod_{i=1}^{p_\eta} \prod_{k=1}^{p_x + p_\theta} \rho_{wik}^{a_{\rho_w} - 1} (1 - \rho_{wik})^{b_{\rho_w} - 1} \times \lambda_y^{a_y' - 1} e^{-b_y' \lambda_y} \times \lambda_v^{a_v - 1} e^{-b_v \lambda_v} \times$$

$$\prod_{k=1}^{p_x} \rho_{vk}^{a_{\rho_v} - 1} (1 - \rho_{vk})^{b_{\rho_v} - 1} \times I[\theta \in C],$$

20

where C denotes the constraint region for θ, which is typically a p_θ-dimensional rectangle. In other applications C can also incorporate constraints between the components of θ.

Realizations from the posterior distribution are produced using standard, single site MCMC. Metropolis updates (Metropolis *et al.*, 1953) are used for the components of ρ_w, ρ_v and θ with a uniform proposal distribution centered at the current value of the parameter. The precision parameters λ_η, λ_w, λ_y and λ_v are sampled using Hastings (1970) updates. Here the proposals are uniform draws, centered at the current parameter values, with a width that is proportional to the current parameter value. Note that we bound the proposal width by not letting it get below 0.5.

In a given application the candidate proposal width can be tuned for optimal performance. However, because of the way the data have been standardized, we have found that a width of 0.2 for the Metropolis updates, and a width of 0.3 times the current parameter value (or 0.5, whichever is larger) for the Hastings updates, works quite well over a fairly wide range of applications. This is an important consideration as we move to develop general software to carry out such analyses.

The resulting posterior distribution estimate for (θ_1, θ_2) is shown in Fig. 10 on the standardized $C = [0,1]^2$ scale. This covers the true values of $\theta = (0.5, 0.5)$ from which the synthetic data were generated.
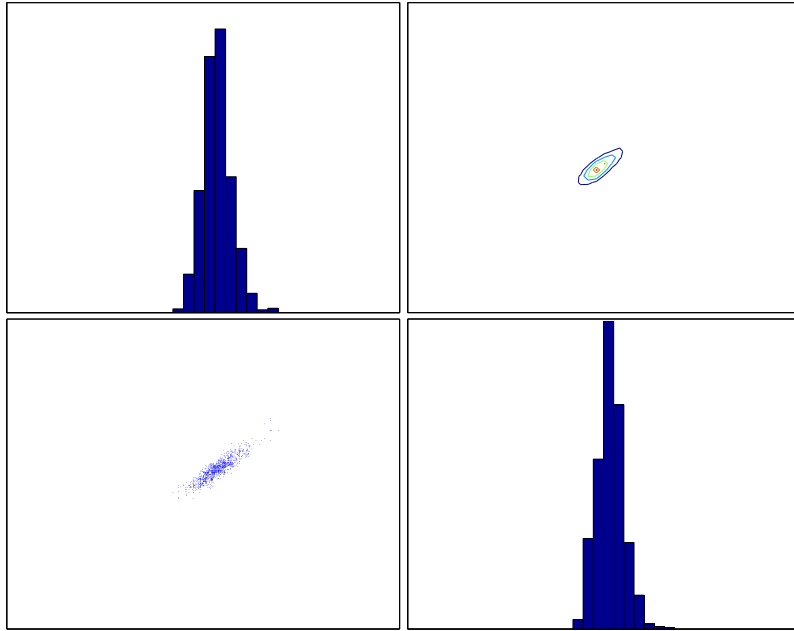

Figure 10: Estimated posterior distribution of the calibration parameters (θ_1, θ_2), which correspond to the detonation energy of the explosive and the yield stress of steel respectively. The true values from which the data were generated are $\theta = (0.5, 0.5)$.

2.5.2 Posterior predictions

As with the pure emulator analysis described in Sec. 2.2, predictions of system behavior can be produced at unobserved input settings $x^\star$. Since

$$
\begin{aligned}
\hat{y}(x^\star) &= \eta(x^\star, \theta) + \delta(x^\star) \\
&= Kw(x^\star, \theta) + Dv(x^\star),
\end{aligned}
$$

we need only produce draws $w(x^\star, \theta)$ and $v(x^\star)$ given a posterior draw of the parameter vector $(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$. Draws of $w(x^\star, \theta)$ and $v(x^\star)$ can then be used to give posterior realizations for the calibrated simulator $\eta(x^\star, \theta)$, the discrepancy term $\delta(x^\star)$, and predictions $\hat{y}(x^\star)$.

These predictions can be produced from standard GP theory. Conditional on the parameter vector $(\lambda_\eta, \lambda_w, \rho_w, \lambda_y, \lambda_v, \rho_v, \theta)$, the reduced data $\hat{z}$, along with the predictions $w(x^\star, \theta)$ and $v(x^\star)$, have the joint distribution

$$
\begin{pmatrix} \hat{z} \\ v(x^\star) \\ w(x^\star, \theta) \end{pmatrix} \sim \mathrm{N}\left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \Sigma_{\hat{z}} & \Sigma_{\hat{z}v^\star} & \Sigma_{\hat{z}w^\star} \\ \Sigma_{v^\star \hat{z}} & \lambda_v I_{p_\delta} & 0 \\ \Sigma_{w^\star \hat{z}} & 0 & \mathrm{diag}(\lambda_w) \end{pmatrix} \right),
$$

where $\Sigma_{\hat{z}v^\star}$ has nonzero elements due to the correlation between $\hat{v}$ and $v(x^\star)$, and $\Sigma_{\hat{z}w^\star}$ has nonzero elements due to the correlation between $(\hat{u}, \hat{w})$ and $w(x^\star, \theta)$. The exact construction of the matrices $\Sigma_{\hat{z}v^\star}$ and $\Sigma_{\hat{z}w^\star}$ is analogous to the construction of Σ_v and Σ_{uw} in Sec. 2.5. Generating simultaneous draws of $v(x^\star)$ and $w(x^\star, \theta)$ is then straightforward using conditional normal rules as is detailed in Sec. 2.2.

The posterior mean estimates for $\eta(x, \theta)$, $\delta(x)$ and their sum, $\hat{y}(x)$, are shown in Fig. 11 for the three input conditions x corresponding to the amount of HE used in each of the three experiments. Also shown are the experimental data records from each of the experiments. Note that the discrepancy term picks up a consistent signal across experiments that varies with time and angle, even though the simulator cannot give variation by angle ϕ.

Figure 12 shows the prediction uncertainty for the inner radius at the photograph times of the three experiments. The figure gives pointwise 90% credible intervals for the inner radius. Here, the prediction for each experiment uses only the data from the other two experiments, making these holdout predictions.
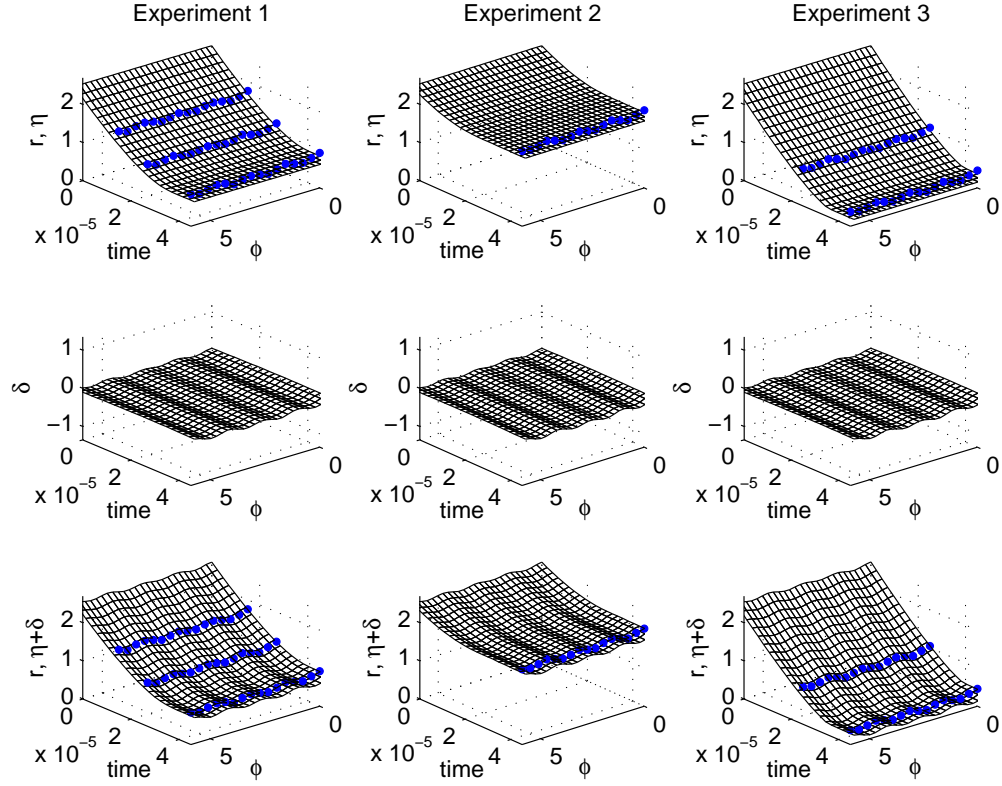
Figure 11: Posterior mean estimates for $\eta(x, \theta)$, $\delta(x)$, and their sum, $\hat{y}(x)$, at the three input conditions corresponding to each of the three experiments. The experimental observations are given by the dots in the figures showing the posterior means for $\eta(x, \theta)$ and $\hat{y}(x)$.

Figure 12: Holdout prediction uncertainty for the inner radius at the photograph times for each experiment. The lines show pointwise 90% credible intervals; the experimental data is given by the dots.

3 Application to HE cylinder experiments

3.1 Experimental setup

Often, analysis of experiments requires that the simulator accurately models a number of different physical phenomena. This is the case with the previous implosion application, which involves imparting energy deposited by an explosive, as well as modeling the deformation of the steel cylinder. The added difficulty of modeling integrated physics effects makes it beneficial to consider additional experiments that better isolate the physical process of interest. The HE cylinder experiment, considered in this section, more cleanly isolates the effects of HE detonation.
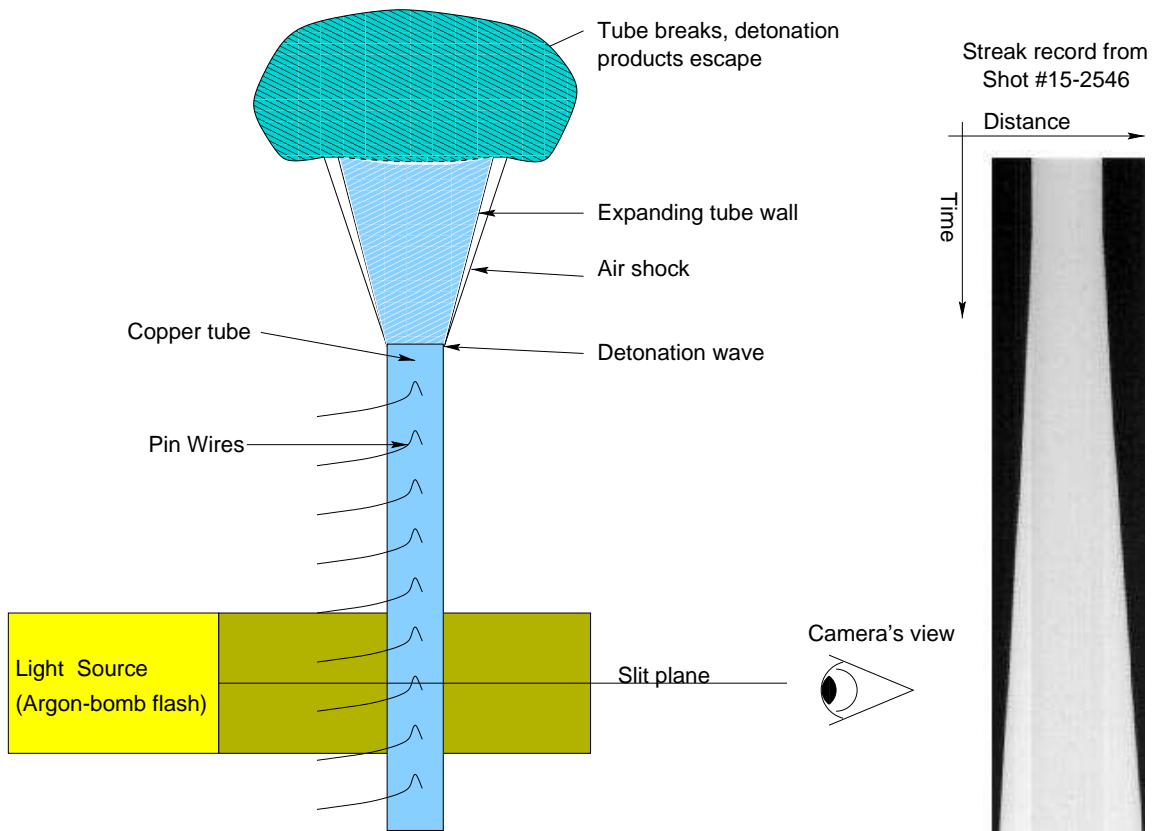
Figure 13: HE cylinder experiment. The HE cylinder is initiated with plane detonation wave, which begins to expand the surrounding copper cylinder as the detonation progresses. This detonation wave moves down the cylinder. Pin wires detect the arrival time of the detonation along the wave, while the streak camera captures the expansion of the detonation wave at a single location on the cylinder.

The cylinder test has become a standard experiment performed on various types of HE at LANL. The standard version of this experiment–depicted in Fig. 13–consists of a thin-walled cylinder of copper surrounding a solid cylinder of the HE of interest. One end of the HE cylinder is initiated with a plane-wave lens; the detonation proceeds down the cylinder of HE, expanding the copper tube via work done by the rapidly increasing pressure from the HE. As the detonation progresses, the copper cylinder eventually fails.

Diagnostics on this experiment generally include a streak camera to record the expansion of the cylinder, and pin wires at regular intervals along the length of the copper cylinder. Each pin wire shorts as the detonation wave reaches its location, sending a signal that indicates time of arrival of the detonation wave. From these arrival times, the detonation velocity of the experiment can be determined with relatively high accuracy. Note the use of copper in this experiment is necessary to contain the HE as it detonates. Since copper is a well-controlled and understood material, its presence will not greatly affect our ability to simulate the experiment.

Details of the experiments are as follows: HE diameter is 1 inch; copper thickness is 0.1 inch; cylinder length is 30 cm; slit location is 19 cm down from where the cylinder is detonated–by this distance, the detonation wave is essentially in steady state. Prior to the experiment, the initial density of the HE cylinder is measured.

3.2   Simulations

Simulations of HE detonation typically involve two major components–the burn, in which the HE rapidly changes phase, from solid to gas, and the equation of state (EOS) for the resulting gas products, which dictates how this gas works on the materials it is pushing against. The detonation velocity, determined by the pin wires, is used to prescribe the burn component of the simulation, moving the planar detonation wave down the cylinder. This empirical approach for modeling the burn accurately captures the detonation for this type of experiment. It is the parameters controlling the EOS of the gaseous HE products that are of interest here.

The EOS describes the state of thermodynamic equilibrium for a fluid (the HE gas products, in this case) at each point in space and time in terms of pressure, internal energy, density, entropy, and temperature. Thermodynamic considerations allow the EOS to be described by only two of these parameters. In this case, the EOS is determined by a system of equations, giving pressure as a function of density and internal energy.

The HE EOS function is controlled by an 8-dimensional parameter vector θ. The first component

θ_1 modifies the energy imparted by the detonation; the second modifies the Gruneisen gamma parameter. The remaining six parameters modify the isentrope lines of the EOS function (pressure-density contours corresponding to constant entropy).

Thus we have 9 inputs of interest to the simulation model. The first, x_1, is the initial density of the HE sample, which is measured prior to the experiment. The remaining $p_\theta = 8$ parameters describe the HE EOS. Prior ranges were determined for each of these input settings. They have been standardized so that the nominal setting is 0.5, the minimum is 0, and the maximum is 1. A 128 run OA-based LH design was constructed over this $p_x + p_\theta = 9$-dimensional input space giving the simulation output shown in Fig. 14. Of the 128 simulation runs, all but two of them ran to completion. Hence the analysis will be based on the 126 runs that completed.
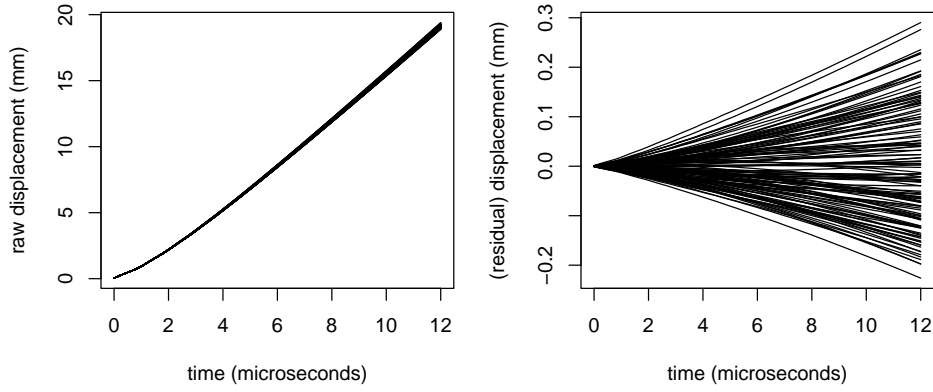

Figure 14: 126 simulated displacement curves for the HE cylinder experiment. Left: simulated displacement of the cylinder where time = 0 corresponds to the arrival of the detonation wave at the camera slit. Right: the residual displacement of the cylinder after subtracting out the pointwise mean of the simulations.

3.3 Experimental observations

Figure 15 shows the experimental data derived from the streak camera from four different HE cylinder experiments. The cylinder expansion is recorded as time-displacement pairs for both the left- and right-hand sides of the cylinder as seen by the streak record. The measured density (in standardized units) for each of the HE cylinders is 0.15, 0.15, 0.33, 0.56 for Experiments 1–4, respectively.

The data errors primarily come from three different sources: determination of the zero dis-
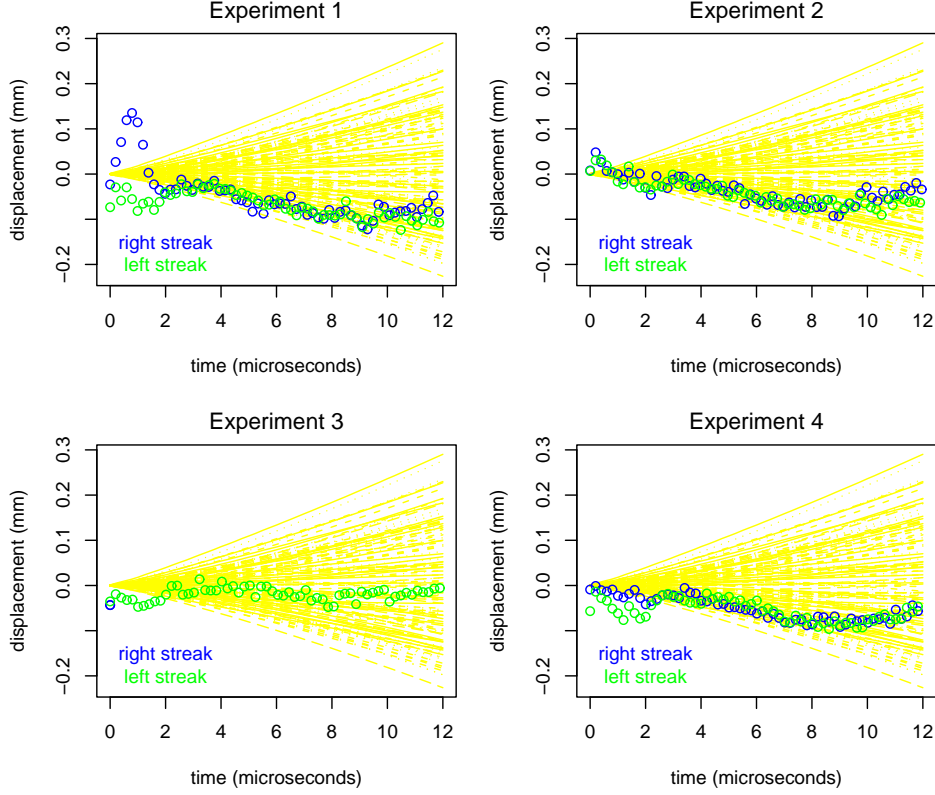
Figure 15: Observed data from 4 HE cylinder experiments. For each streak camera image, points corresponding to displacement are recorded over time. Separate displacements are recorded for the left and right side of the cylinder. Here, the mean of the simulated displacements has been subtracted from the data. Note that Experiment 3 is missing the displacement from the right-hand cylinder expansion. The light lines in the background of each figure show the 126 simulated displacements.

placement level, causing a random shift in the entire data trace; replicate variation due to subtle differences in materials used in the various experiments–modeled as time correlated Gaussian errors; and jitter due to the resolution of the film. After substantial discussion with subject matter experts, we decided to model the precision W_i for Experiment i as

$$W_i = \left(\sigma_0^2 \mathbf{1}\mathbf{1}^T + \sigma_a^2 R_y + \sigma_j^2 I\right)^{-1},$$

where the variances σ_0^2, σ_a^2, and σ_j^2 and correlation matrix R_i have been elicited from experts. Note that the model allows for a parameter λ_y to scale the complete data precision matrix W_y. Because of the ringing effects in the displacement observations at early times, we take only the data for

which $2.5\mu s \leq t \leq 12.0\mu s$.

3.4 Analysis and results

We model the simulation output using a principal component basis (Fig. 16) derived from the 126 simulations over $0\mu s \leq t \leq 12\mu s$. Eight basis functions (also shown in Fig. 16) are used to
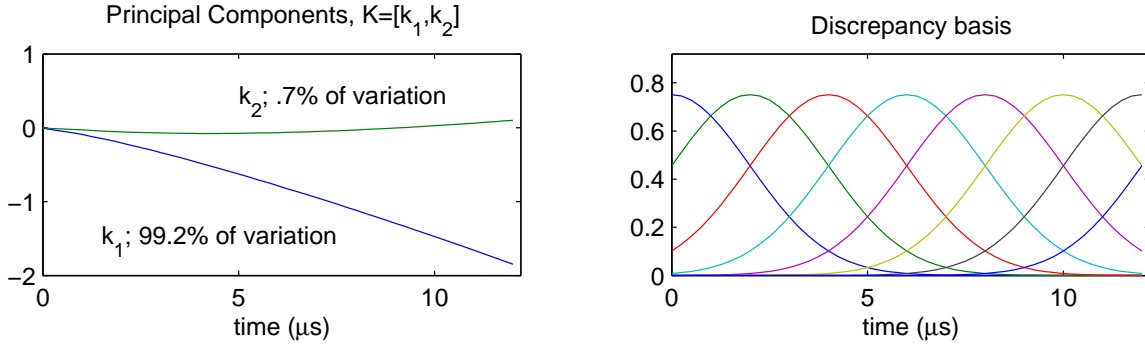

Figure 16: Principal component basis (left) and the kernel-based discrepancy basis (right). The discrepancy uses independent copies of the kernel basis shown in the right-hand plot for the left and right streaks. Here $p_\eta = 2$ and $p_v = 2 \cdot 8$.

determine the discrepancy $\delta(x)$ as a function of time for each side of the cylinder expansion seen in the streak record.

The fitted model for the simulation output allows us to explore the sensitivity of the simulated expansion to the 9 input parameters. Figure 17 shows the posterior mean of the simulator output $\eta(x_1, \theta)$ as one of the input parameters is varied while the remaining parameters are held at their nominal value of 0.5. From the figure it is apparent that x_1–HE density, θ_1–detonation energy, and θ_3–the first isentrope parameter, have the most effect on the simulated expansion.

The MCMC output resulting from sampling the posterior distribution of the full model (12) allows us to construct posterior realizations for the calibration parameters θ, the discrepancy process $\delta(x)$, and predictions of the cylinder expansion at general input condition x_1. The estimated 2-dimensional marginal distributions for $\pi(\theta|y)$ are shown in Fig. 18. The lines show estimated 90% HPD regions for each margin. Recall the prior is uniform for each component of θ.

For this application, the posterior distribution for the discrepancy process is tightly centered at 0 for both the left- and right-hand expansion. Hence the experimental error term ϵ adequately accounts for the simulations and experimental data.
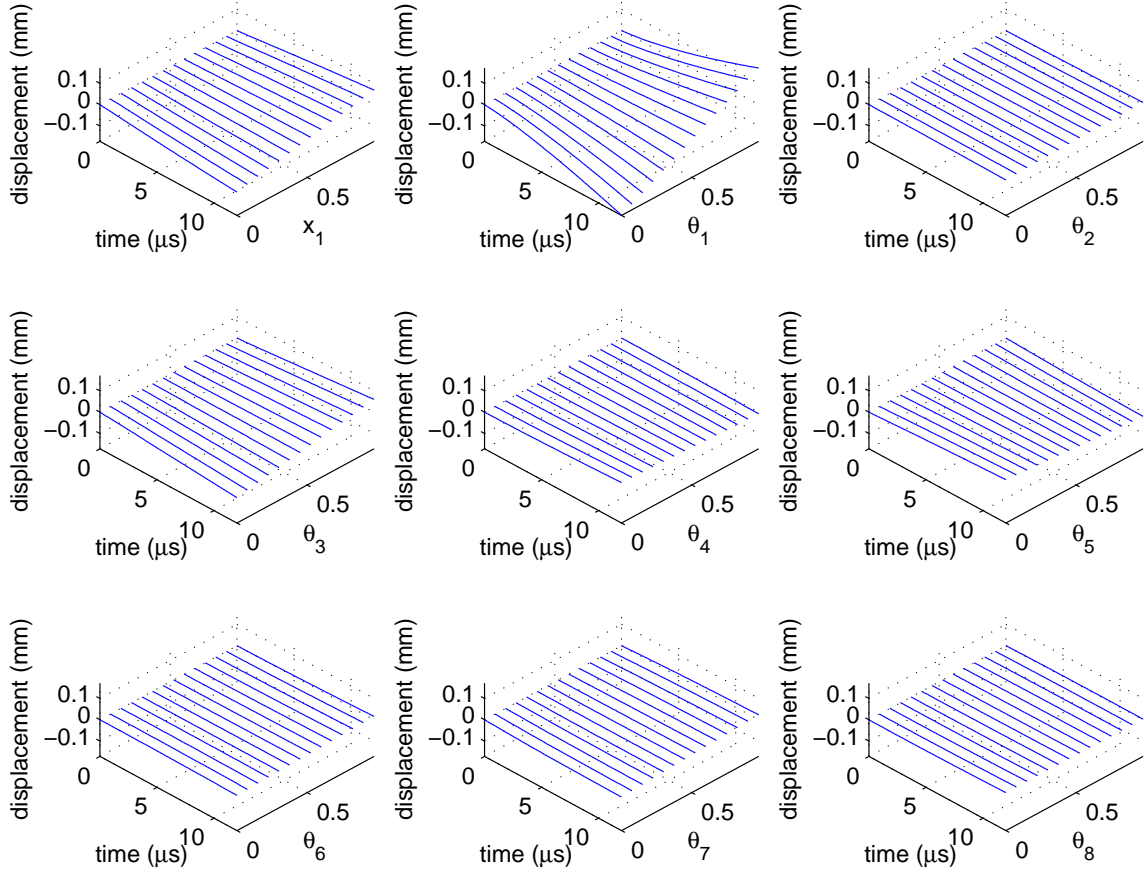
Figure 17: Estimated sensitivites of the simulator output from varying a single input while keeping the remaining eight inputs at their nominal value.

Posterior predictions for the cylinder expansion are given in Fig. 19 at input conditions x_1 corresponding to the four experiments used in this analysis. Also shown are the experimental observations. Note that experiments 1 and 2 are essentially replicates. Thus they inform about the magnitude of the experiment to experiment variation which is captured in the precision parameter λ_y.

4 Discussion

The modeling approach described in this paper has proven quite useful in a number of applications at LANL. Application areas include shock physics, material science, engineering, cosmology, and particle physics.

The success of this approach depends, in large part, on whether or not the simulator can be
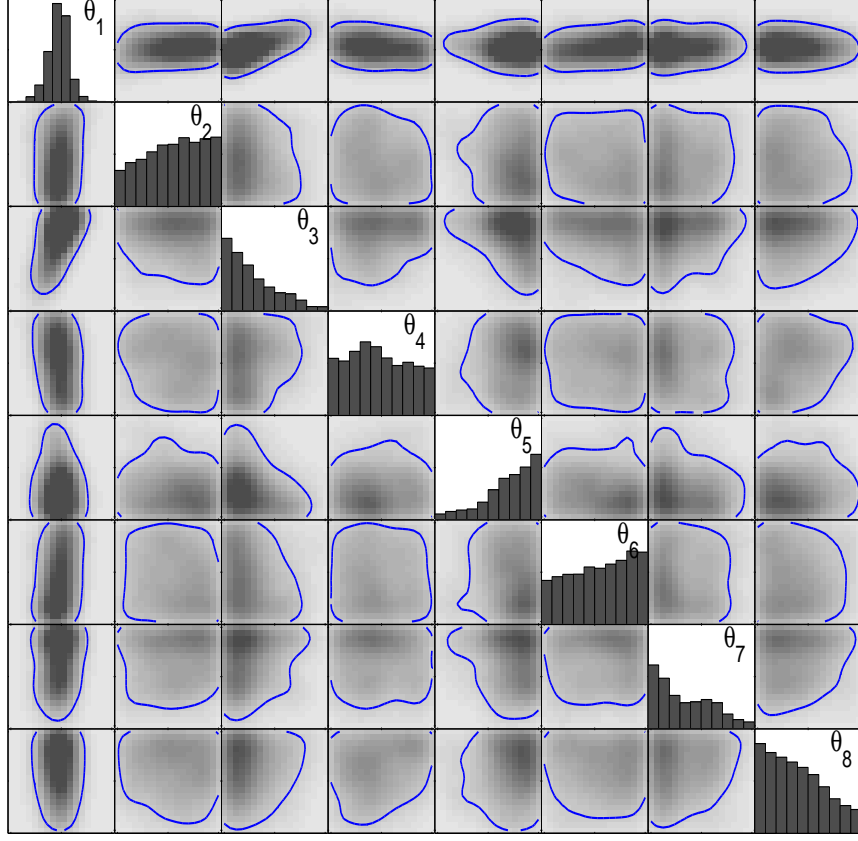
Figure 18: Two-dimensional marginals for the posterior distribution of the eight EOS parameters. The solid line gives the estimated 90% HPD region. The plotting regions correspond to the prior range of $[0, 1]$ for each standardized parameter.

efficiently represented with the GP model on the basis weights $w_i(x, \theta)$, $i = 1, \ldots, p_\eta$. This is generally the case for highly forced systems–such as an implosion–which are dominated by a small number of modes of action. This is apparent in the principal component decomposition, which partitions nearly all of the variance in the first few components. These systems also tend to exhibit smooth dependence on the input settings. In contrast, more chaotic systems seem to be far less amenable to a low-dimensional description such as the PC-basis representations used here. Also, system sensitivity to even small input perturbations can look almost random, making it difficult to construct a statistical model to predict at untried input settings. We therefore expect that an alternative approach is required for representing the simulator of a less forced, more chaotic system.

Finally, we note that the basic framework described here does lead to issues that require careful consideration. The first is the interplay between the discrepancy term $\delta(x)$ and the calibration
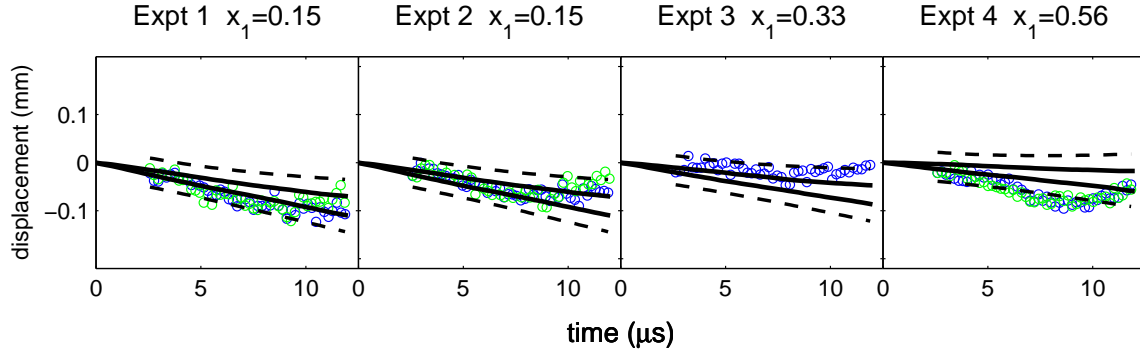
Figure 19: Posterior distribution for the cylinder expansion at the input conditions corresponding to the four experiments. The solid lines show pointwise 90% credible intervals for $\eta(x_1, \theta)$; the dashed lines give pointwise 90% credible intervals for a new measurement $y^*(x_1)$. The circles show the observed streaks from the experiments.

parameters θ; this is noted in the discussion of Kennedy and O'Hagan (2001), as well as in a number of other articles (Higdon *et al.*, 2004, Bayarri *et al.*, 2005, and Loeppky *et al.*, 2005). The basic point here is that if there is a substantial discrepancy, then its form will affect the posterior distribution of the calibration parameters. The second is the issue of extrapolating outside the range of experimental data. The quality of such extrapolative predictions depends largely on the trust one has for the discrepancy term $\delta(x)$–is what's learned about the discrepancy term at tried experimental conditions $x_1, \ldots, x_n$ applicable to a new, possibly far away, condition x^*? Third, and last, we note that applications that involve a number of related simulation models require additional modeling considerations to account for the relationship between the various simulation models. See Kennedy and O'Hagan (2000) and Goldstein and Rougier (2005) for examples.

Acknowledgments

References

Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C. H. and Tu, J. (2005). A framework for validation of computer models, *Technical report*, Statistical and Applied Mathematical Sciences Institute.

Bengtsson, T., Snyder, C. and Nychka, D. (2003). Toward a nonlinear ensemble filter for high-dimensional systems, *Journal of Geophysical Research* **108**: 1–10.

Berliner, L. M. (2001). Monte carlo based ensemble forecasting, *Statistics and Computing* **11**: 269–275.

Goldstein, M. and Rougier, J. C. (2005). Probabilistic formulations for transferring inferences from mathematical models to physical systems, *SIAM Journal on Scientific Computing* **26**: 467–487.

Hanson, K. M. and Cunningham, G. S. (1999). Operation of the Bayes inference engine, *in* W. von der Linden (ed.), *Maximum Entropy and Bayesian Statistics*, pp. 309–318.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**: 97–109.

Hegstad, B. K. and Omre, H. (2001). Uncertainty in production forecasts based on well observations, seismic data and production history, *Society of Petroleum Engineers Journal* pp. 409–424.

Higdon, D., Kennedy, M., Cavendish, J., Cafeo, J. and Ryne, R. D. (2004). Combining field observations and simulations for calibration and prediction, *SIAM Journal of Scientific Computing* **26**: 448–466.

Higdon, D. M., Lee, H. and Holloman, C. (2003). Markov chain monte carlo-based approaches for inference in computationally intensive inverse problems, *in* J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith and M. West (eds), *Bayesian Statistics 7. Proceedings of the Seventh Valencia International Meeting*, Oxford University Press, pp. 181–197.

Kaipio, J. P. and Somersalo, E. (2004). *Statistical and Computational Inverse Problems*, Springer, New York.

Kaipio, J. P., Kolehmainen, V., Somersalo, E. and Vauhkonen, M. (2000). Statistical inversion and monte carlo sampling methods in electrical impedance tomography, *Inverse Problems* **16**(5): 1487–1522.

Kao, J., Flicker, D., Ide, K. and Ghil, M. (2004). Estimating model parameters for an impact-produced shock-wave simulation: optimal use of partial data with the extended Kalman filter, *Journal of Computational Physics* **196**: 705–723.

Kennedy, M. and O'Hagan, A. (2000). Predicting the output from a complex computer code when fast approximations are available, *Biometrika* **87**: 1–13.

Kennedy, M. and O'Hagan, A. (2001). Bayesian calibration of computer models (with discussion), *Journal of the Royal Statistical Society (Series B)* **68**: 425–464.

Kern, J. (2000). Bayesian process-convolution approaches to specifying spatial dependence structure, Ph.D. Thesis, Institute of Statistics and Decision Sciences, Duke University.

Loeppky, J., Bingham, D., Sacks, J. and Welch, W. J. (2005). Based estimation of claibration and tuning parameters in a possibly biased computer model, *Technical report*, University of British Columbia.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E. (1953). Equations of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1091.

Nakhleh, C., Higdon, D., Allen, C. and Kumar, V. (2005). Bayesian reconstruction of particle beam phase space from low dimensional data, *Technical Report LA-UR-05-5897*, Los Alamos National Laboratory.

Neddermeyer, S. (1943). Collapse of hollow steel cylinders by high explosives, *Los Alamos Report LAR-18*, Los Alamos National Laboratory.

Oakley, J. and O'Hagan, A. (2004). Probabilistic sensitivity analysis of complex models, *Journal of the Royal Statistical Society (Series B)* **66**: 751–769.

Ramsay, J. O. and Silverman, B. W. (1997). *Functional Data Analysis*, Springer, New York.

Sacks, J., Welch, W. J., Mitchell, T. J. and Wynn, H. P. (1989). Design and analysis of computer experiments (with discussion), *Statistical Science* **4**: 409–423.

Tang, B. (1993). Orthogonal array-based Latin hypercubes, *Journal of the American Statistical Association* **88**: 1392–1397.